

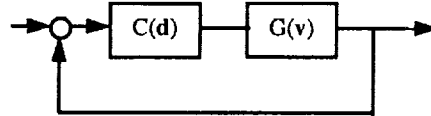
511-61
207605
p. 11

The purpose of this research is to create a method of finding practical, robust control laws. The robustness of a controller is judged by Stochastic Robustness metrics and the level of robustness is optimized by searching for design parameters that minimize a robustness cost function.

PRECEDING PAGE BLANK NOT FILMED

PAGE 98 INTENTIONALLY BLANK

Review of the Stochastic Robustness Metric



$$P(C) = \int_{\mathbf{v}} H(C, \mathbf{v}) \text{pr}(\mathbf{v}) d\mathbf{v}$$

$$J(C) = \text{fn}(P_1(C), P_2(C), \dots)$$

Given $C = C(d)$ find d^* :

$$J(C(d^*)) = \min_{d \in D} (J(C(d)))$$

Estimate by Monte Carlo Analysis

$$\hat{P}(C) = \frac{1}{N} \sum_i H_i(C, \mathbf{v}_i) \quad \mathbf{v}_i \text{ from } \text{pr}(\mathbf{v})$$

$$\sigma_{\hat{P}_j} = \sqrt{\frac{P - P^2}{N}}$$

$$\hat{J}(C) = \text{fn}(\hat{P}_1(C), \hat{P}_2(C), \dots)$$

σ_j from Bootstrap

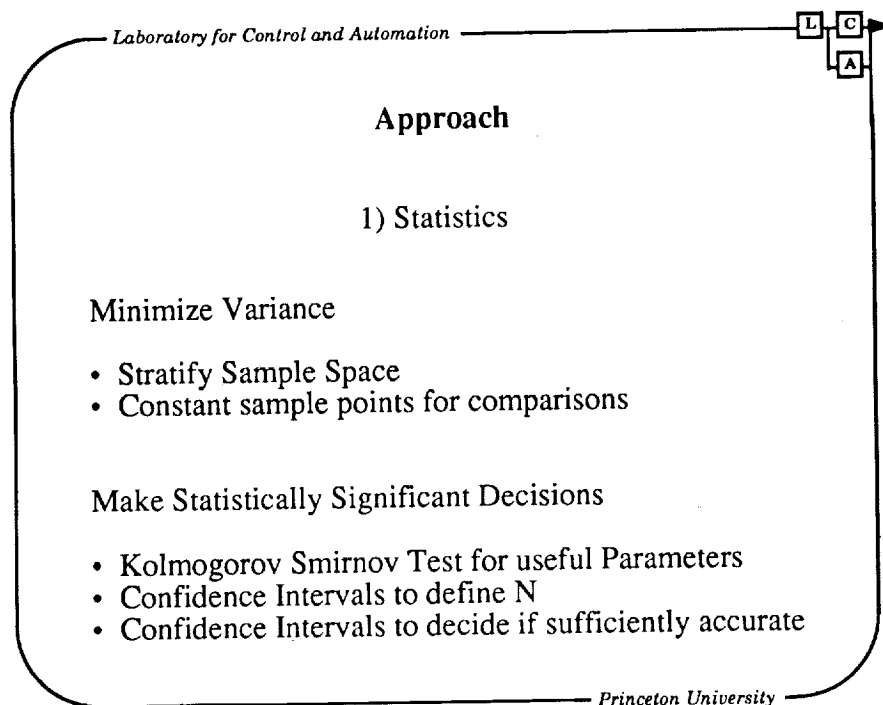
Princeton University

Given the expected variation of the plant parameters, a Stochastic Robustness metric characterizes a compensator by giving the probability that the compensator will fail to perform acceptably. The definition of what is unacceptable is left to the designer but will normally include such features as instability and slow response time. To calculate the probability of unacceptability, P , the indicating function, $H(C, \mathbf{v})$ must be integrated over the space of expected parameter variations. H is a function of both the compensator, C and the plant parameter values, \mathbf{v} . H equals one when the metric is violated and zero otherwise.

Normally, more than one metric will be of importance in a given application. In such a case it may be necessary to make a trade-off between the metrics. The trade-off can be formalized by combining the probabilities into a scalar cost function, J . Weights within the cost function can then be used to reflect the importance to the designer of each metric.

Once J is defined, the task is to find the set of plant design parameters, \mathbf{d} , to minimize J .

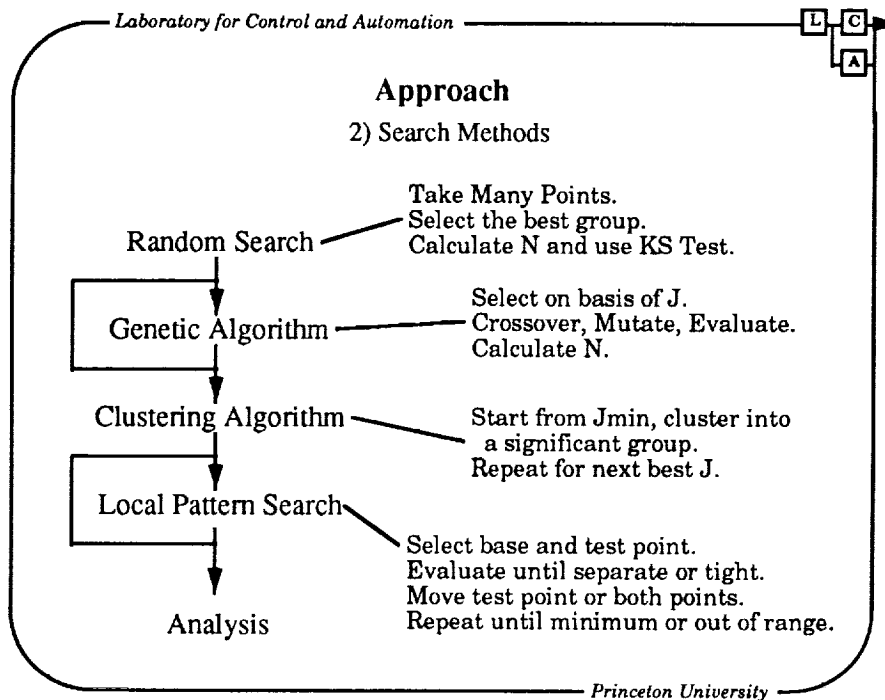
This task is hindered by the fact that it is normally impossible to evaluate the probabilities analytically. An alternative evaluation method is to use Monte Carlo Analysis; this has the disadvantage that errors can be expected in the estimate of P . The expected error reduces as the inverse of the square root of the number of evaluations. There is therefore a trade-off between the accuracy and of the evaluations and the computation time.



The approach to finding a stochastic global optimization method has two main thrusts. The first is to understand the statistical effects of the Monte Carlo Analysis and exploit them to reduce the number of evaluations necessary. The second approach is to identify suitable search algorithms.

The variability in the estimates of P has been reduced significantly by stratifying the sample space and by using the same sample points when comparing two compensators. An understanding of these statistical mechanisms has allowed a significant reduction in the number of evaluations which must be carried out to compare two compensators.

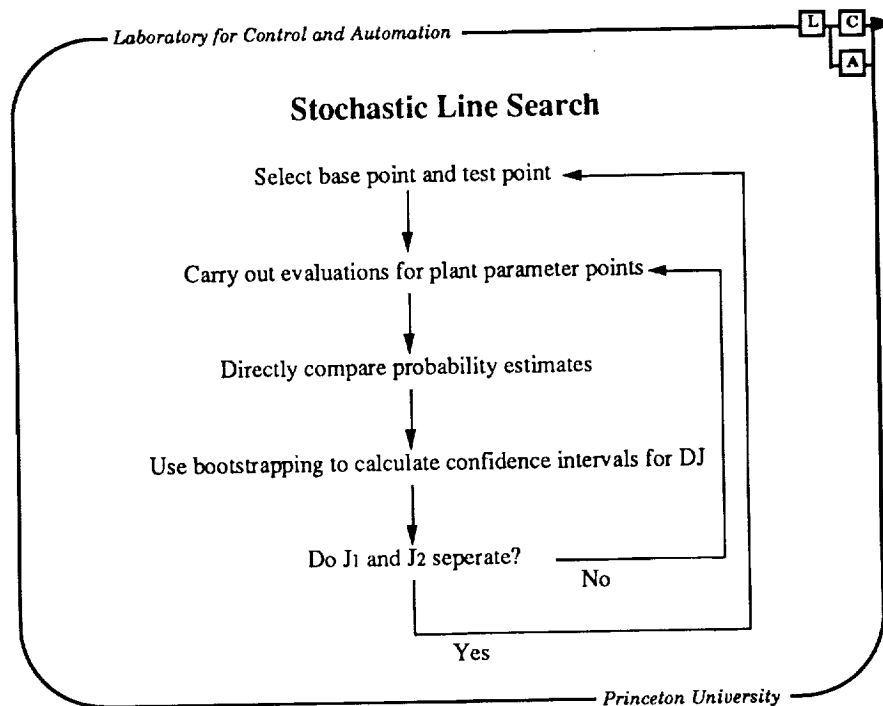
By using the Kolmogorov-Smirnov test it is possible to identify parameters that have a significant effect on J , allowing the search to concentrate on these parameters. The establishment of confidence intervals on the estimates of P provide a basis for making statistically significant search decisions and also to fix the number of further evaluations that must be required if the results are not yet statistically significant.



A wide range of modern search methods were screened for their possible use in searching stochastic space. The most efficient method combines the best qualities of several different methods.

The proposed search method begins by taking a broad, completely random, search across the design space. A few evaluations are made at each point and the best points are then presented as the starting population for a genetic algorithm. The genetic algorithm carries out the bulk of the search and later will be described in detail.

The result of the genetic algorithm is a set of candidate solutions, most of which should be close to the global minimum. A clustering algorithm is then used to identify groups of good solutions and a local line search is carried out from the centroid of each cluster.



The line search is based on a pattern search with additional logic to deal with the uncertainties introduced by the Monte Carlo Evaluation.

The search moves along the line, comparing two points at a time. A set amount of Monte Carlo Evaluations are carried out and then a decision is made as to where along the line the next evaluations should be made. The decision is based on an estimate of the likely error in J . If the errors are relatively small then we can be confident that there is a true difference between the compensators and a new search point can be chosen. If the error is relatively large, more evaluations need to be carried out.

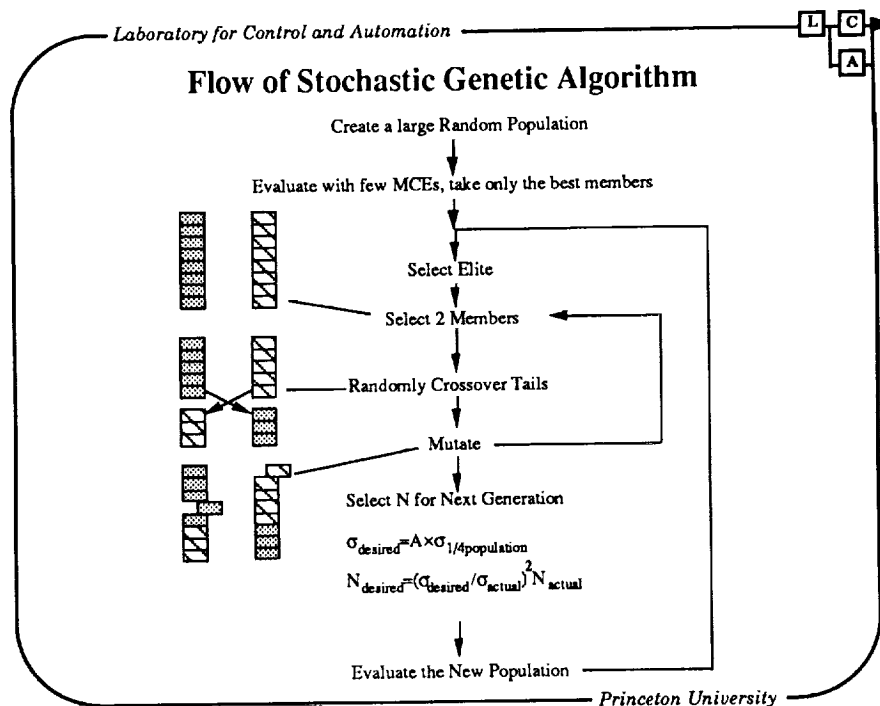
This search method has been implemented, and is effective in finding the minimum along a line in design parameter space.

Global Optimization by Genetic Algorithms

- GAs are Partially Randomized, therefore suitable for SRA.
- Efficient: Exponential Replication of Good Parameter Values.
- Little Previous work with noisy functions.
- No work with Monte Carlo Optimization. $\sigma_{p_j} = \sqrt{p - p^2} / N$

Princeton University

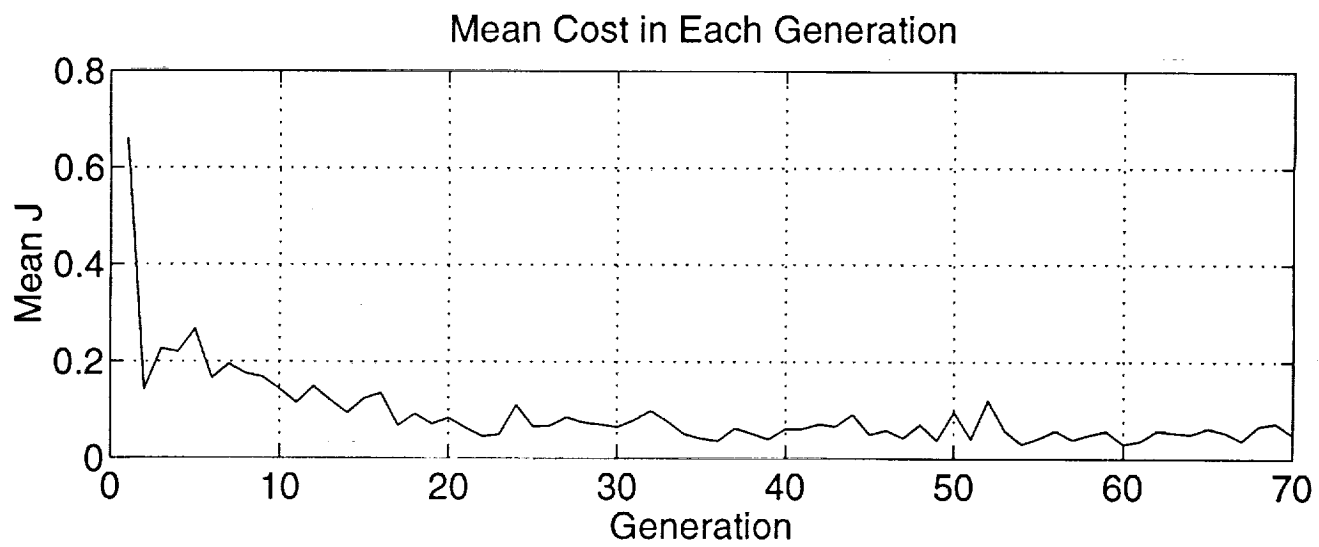
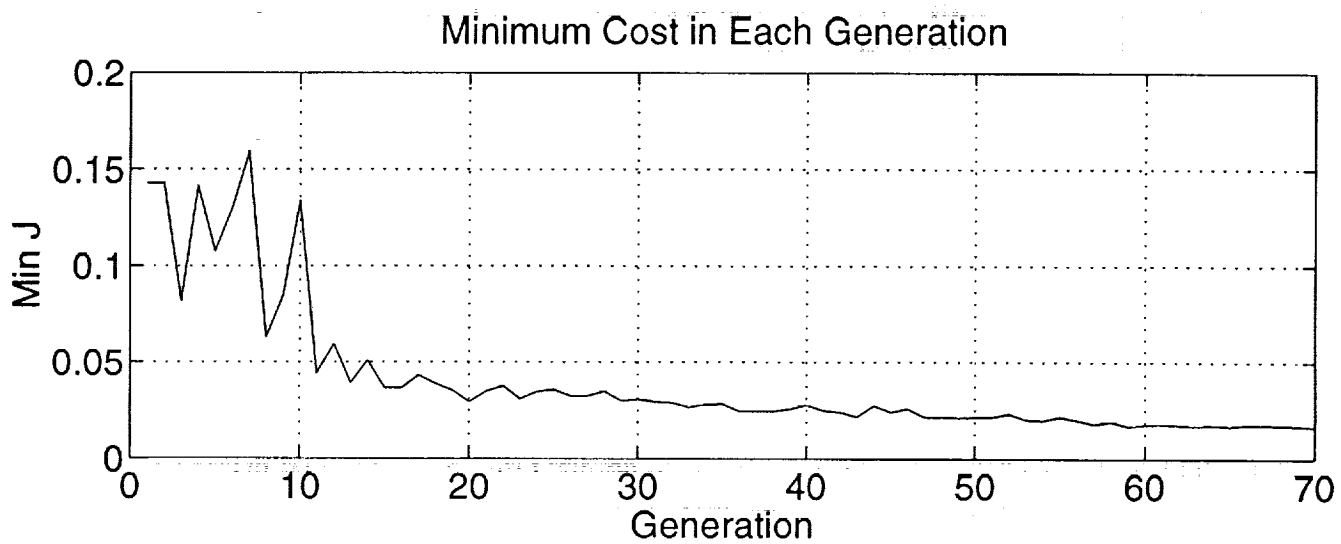
Genetic Algorithms (GAs) were chosen as the main global optimization method. These algorithms have several attributes that make them well suited to searching a stochastic space. They rely on a partially randomized comparison of many points and are therefore insensitive to errors due to Monte Carlo Evaluation and they process information efficiently. However, little previous work has been done in using GAs to optimize noisy functions. This work must be carried out before using GAs for the synthesis of robust control.



The Stochastic Genetic Algorithm (SGA) is currently being researched. The basic structure of the SGA is shown above. The SGA is similar to normal GAs except for 3 points:

- 1) The search begins with a random search, using a few Monte Carlo Evaluations at each point, and using a small proportion of the random points as the initial population to 'kick-start' the SGA.
- 2) The Kolmogorov-Smirnov test is used to determine which design parameters are most important in affecting J. These parameters are used to cluster the best members of the population to form one averaged member. This is passed as an elite member into the next generation.
- 3) The number of evaluations per point, N, is fixed before each set of Monte Carlo Evaluations. This is done by comparing the expected error in the estimated cost of the best member of the population with the mean difference between the costs of the rest of the population. The ratio of the error allowed in the estimate to the difference in the cost of the population can be varied to improve the performance of the search. Here, this parameter is referred to as "A".

The next graphs show the results of a typical run of the SGA. The first graph shows the values of J for the best member in the population of each generation as the population evolves to a low value of J. The second graph shows the mean value for J for each generation.



Parameters to be Tuned

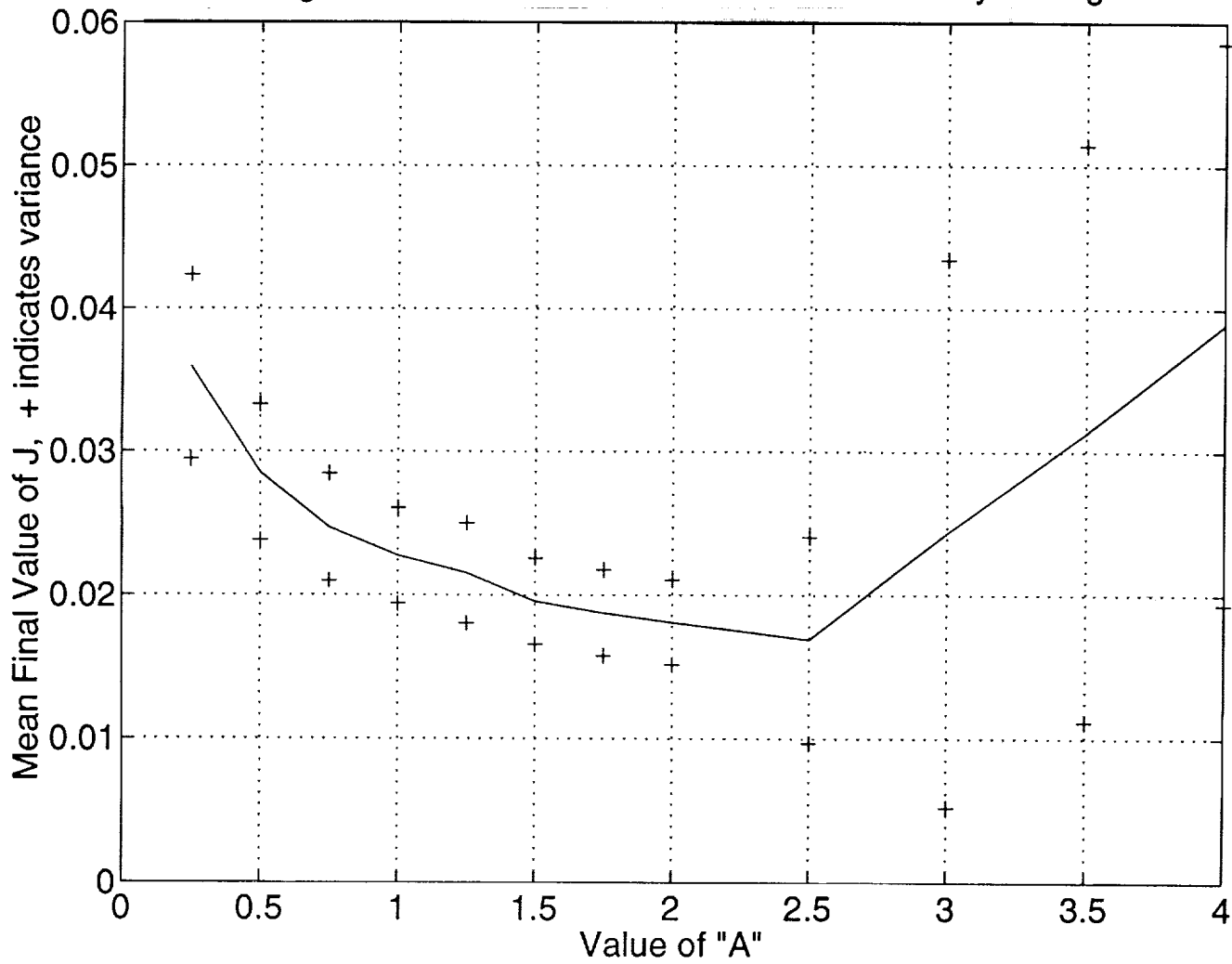
- Number of members in random population
- Number of initial evaluations
- Number of members in genetic population
- Number of elite members passed down
- Value of A to fix level expected error
- Number of crossover points
- Probability of crossover
- Number of mutations
- Probability of mutation
- Degree of mutation

Princeton University

Within the SGA there are several parameters that must be carefully chosen to ensure that the search is efficient. These parameters are being tuned by running the SGA repeatedly on a test function, adjusting the parameter, and running the SGA again.

The next graph shows the effect of changing the value of A. Here the SGA was run 150 times for each of 12 different values of A. At low values of A, few evaluations are carried out per point and the SGA does not have information of sufficient quality to converge well; with high values of A the information is of higher quality than needed and the computational effort would be better spent searching more points. The optimum value is between 2 and 3. With $A = 3$ the performance is occasionally very good but on average the result is mediocre. With $A = 2.5$ the performance will on average be the best but there is a relatively wide variability. With $A = 2$ the average performance is not quite so good but the search is more robust; the variability is less and the search is less likely to result in a poor outcome.

Change in the Mean Performance as the Accuracy Changes



Future Work

- Complete the tuning of the Genetic Algorithm.
- Combine the Genetic Algorithm with the Line Search.
- Test the method on a real-world control problem.

Future work will complete the tuning of the SGA and combine it with the line search. The overall algorithm will then be tested against real world control synthesis problems.

